Assignment problem

The general idea of B&B is a BFS-like search for the optimal solution, but not all nodes get expanded (i.e., their children generated). Rather, a carefully selected criterion determines which node to expand and when, and another criterion tells the algorithm when an optimal solution has been found.

- Branch and Bound is a technique that is widely used for speeding up a backtracking algorithm.
- "backtracking with branch and bound".
- We have a recursive algorithm that tries to build a solution part by part, and when it gets into a dead end, then it has either built a solution or it needs to go back (backtrack) and try picking different values for some of the parts.
- We check whether the solution we have built is a valid solution only at the deepest level of recursion –when we have all parts picked out.
- Branch and bound says that sometimes, we can notice that after building only a partial solution there is no need to go any deeper because we are heading into a dead end.

Assignment Problem

- **Input: n jobs, n employees, and an n x n matrix A where A_{ij} be the cost if person i performs job j.**
- **Problem: find a one-to-one matching of the n employees to the n jobs so that the total cost is minimized.**

	1	2	3
a	4	7	3
b	2	6	1
c	3	9	4

$$\begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline a & 4 & 7 & 3 \\ b & 2 & 6 & 1 \\ c & 3 & 9 & 4 \end{array}$$

If we allot task 1 to agent a, task 2 to agent b, and task 3 to agent c, then our total cost will be $4 + 6 + 4 = 14$, while if we allot task 3 to agent a, task 2 to agent b, and task 1 to agent c, the cost is only $3 + 6 + 3 = 12$. In this particular example, the reader may verify that the optimal assignment is $a \to 2$, $b \to 3$, and $c \to 1$, whose cost is $7 + 1 + 3 = 11$.

Example

	T1	T2	T3
P1	5	8	4
P2	3	7	2
P3	4	10	5

If

Persons	Task	Cost	Total Cost is 17
p1	T3	4	
p2	T1	3	
p3	T2	10	

If

Persons	Task	Cost	Total Cost is 16
p1	T2	8	
p2	T1	3	
p3	T3	5	

Optimal assignment for this is:

Persons	Task	Cost	Total Cost is 14
p1	T2	8	
p2	T3	2	
p3	T1	4	

Cost matrix:

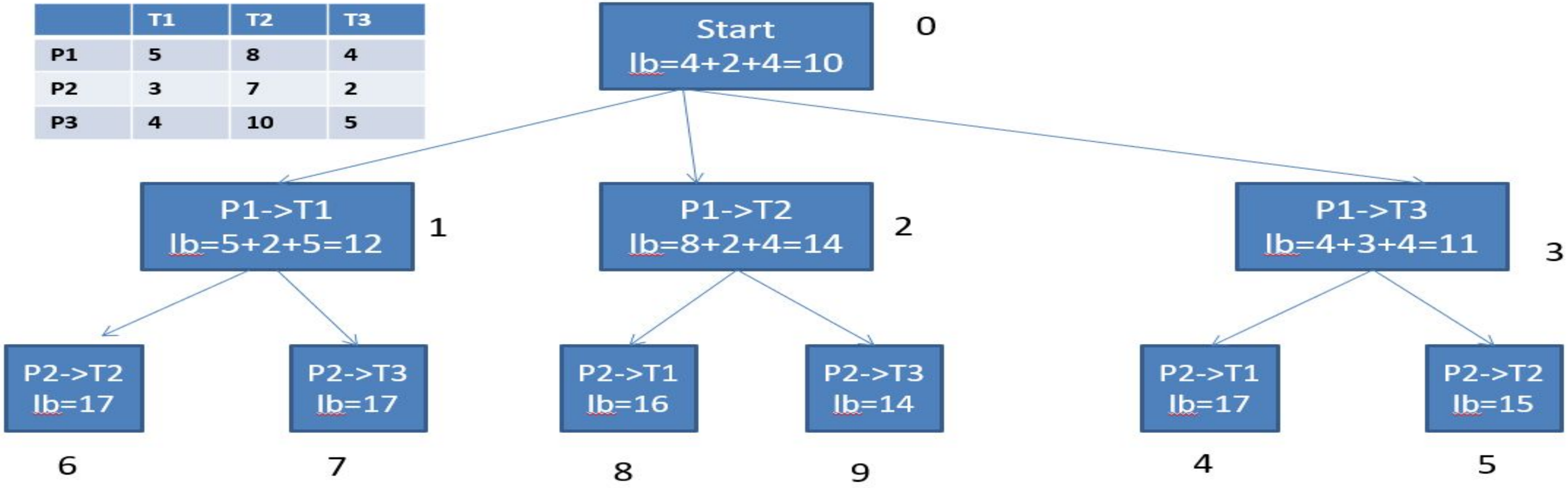
	T1	T2	T3
P1	5	8	4
P2	3	7	2
P3	4	10	5

- Cost of any solution could not be less than the lower bound.
- Lower bound= Sum of the minimum value from each row
- For this example it is lb= 4+2+4=10

Steps:

- Assign T1 to P1, T2 to P1, T3 to P1 and calculate lb for these three possibilities.
- Select the node having minimum lb.
- Give the nodes numbers according to their visit.
- If the lower bounds of the nodes exceed the lb of node labelled 1, explore it and calculate lower bound.
- After examining each leaves in these order, select the assignment with optimal lower bound.

	T1	T2	T3
P1	5	8	4
P2	3	7	2
P3	4	10	5



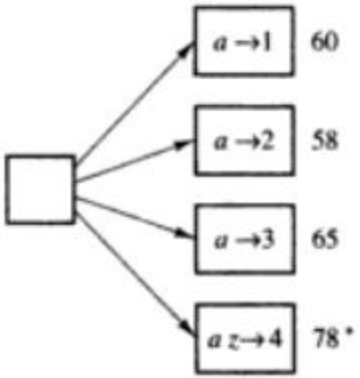
So the final assignment solution would be
P1->T2 (8)
P2->T3 (2)
P3->T1 (4)
Cost is : 14

Suppose we have to solve the instance whose cost matrix is shown in Figure 9.13. To obtain an upper bound on the answer, note that $a \to 1$, $b \to 2$, $c \to 3$, $d \to 4$ is one possible solution whose cost is $11 + 15 + 19 + 28 = 73$. The optimal solution to the problem cannot cost more than this. Another possible solution is $a \to 4$, $b \to 3$, $c \to 2$, $d \to 1$ whose cost is obtained by adding the elements in the other diagonal of the cost matrix, giving $40 + 13 + 17 + 17 = 87$. In this case the second solution is no improvement over the first. To obtain a lower bound on the solution, we can argue that whoever executes task 1, the cost will be at least 11; whoever executes task 2, the cost will be at least 12, and so on. Thus adding the smallest elements in each column gives us a lower bound on the answer. In the example, this is $11 + 12 + 13 + 22 = 58$. A second lower bound is obtained by adding the smallest elements in each row, on the grounds that each agent must do something. In this case we find $11 + 13 + 11 + 14 = 49$, not as useful as the previous lower bound. Pulling these facts together, we know that the answer to our instance lies somewhere in $[58..73]$.

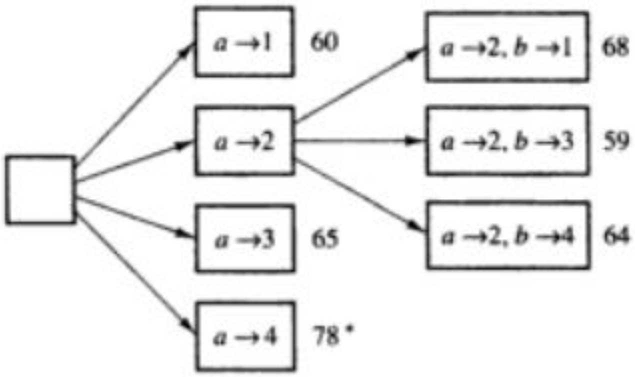
	1	2	3	4
a	11	12	18	40
b	14	15	13	22
c	11	17	19	23
d	17	14	20	28

Figure 9.13. The cost matrix for an assignment problem

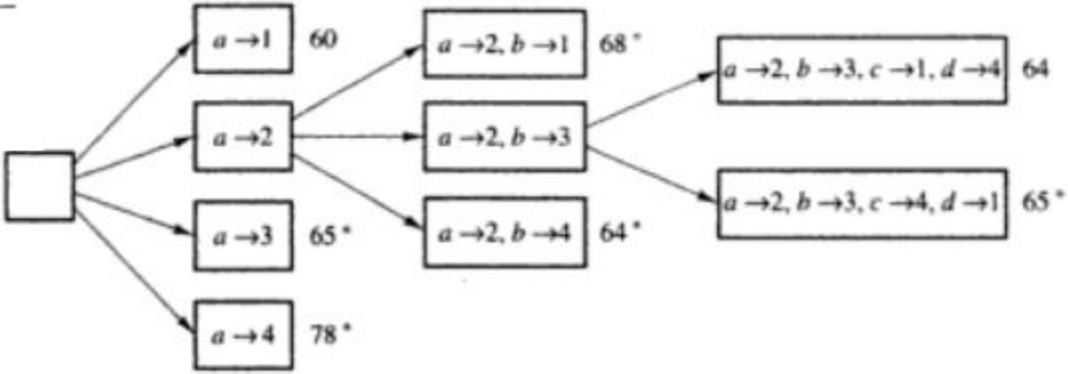
	1	2	3	4
a	11	12	18	40
b	14	15	13	22
c	11	17	19	23
d	17	14	20	28



	1	2	3	4
a	11	12	18	40
b	14	15	13	22
c	11	17	19	23
d	17	14	20	28



	1	2	3	4
a	11	12	18	40
b	14	15	13	22
c	11	17	19	23
d	17	14	20	28



	1	2	3	4
a	11	12	18	40
b	14	15	13	22
c	11	17	19	23
d	17	14	20	28